Summary on
Neuro.ZERO: A Zero-Energy Neural Network Accelerator for Embedded Sensing and Inference Systems

Lohitanvita Rompicharla

RUID: 211009876

Department of Electrical and Computer Engineering

Objective

The advancement in technology has brought machine learning and recently deep learning models into our daily life. Starting with speech recognition, email spamming, facial recognition to now creating chatbots, virtual assistants, and embedding deep learning models to smartphones has become a trend of achievement. The development in leveraging deep learning neural networks has paved a way for deep neural network accelerators, which provide the basic architecture to incorporate faster execution of DL models in smart devices like mobile phones. But the performance of the accelerators is affected by major practical limitations, such as consuming a lot of power, degradation in inference accuracy due to lack of adaptability, lack of standardization, and unaffordability. Therefore, to overcome these limitations, a co-processor architecture called Neuro.Zero is introduced that aims at achieving higher run-time performance of DNN on resource-constrained MCU-based system by using a low-power energy harvesting MCU as an accelerator.

Introduction

The novel approach Neuro.Zero architecture used for improving run-time performance in DNN accelerators is based on two micro-controller units namely a battery-powered main MCU and a batteryless (energy-harvesting) accelerator MCU. The battery-powered accelerator executes a scaled-down inference task whereas the batteryless accelerator runs on the main MCU and does not draw power from the main system, hence this zero-energy accelerator is named Neuro.Zero. To achieve its performance goal, Neuro.Zero follows four acceleration modes, they are, 1) extended interference, 2) expedited inference, 3) ensemble inference, and 4) latent training. Also, for enabling these four modes, a set of numerical algorithms, 1) energy and intermittence-aware algorithms, and 2) fast and high-precision adaptive fixed-point arithmetic is practiced that facilitate a larger size network by following parallel execution and updating DNN weights by online training. In terms of design as well, the main factors such as CPU performance, power consumption, and cost are immensely satisfied by choosing MCU compared to FPGA or SoC for the harvester. Since the main MCU is only connected to the battery and takes less time for charge-discharge cycles, seamless execution of tasks with an increase in performance due to the energy-harvesting accelerator is observed. Also, this increase in accuracy and speedup outweighs the extra cost spent on MCU addition.

Working

As discussed in the introduction section, Neuro.Zero architecture consists of two microcontrollers, one as the main MCU and the other as an accelerator that operates at a low-power (118µA – 1.8mA) supplied from an energy-harvester. Apart from MCUs, the hardware platform consists of sensors (microphone sensor, camera), Memory space (FRAM: Ferroelectrics for Digital Information Storage and Switching), and a capacitor for energy storage that accelerates the system. Neuro.Zero is dependent on the compile-time tool and a run-time system. The compile-time tool helps in generating two DNNs for both MCUs by training the architecture with the use of a training dataset and acceleration mode. The run-time system executes these two generated DNNs by managing the coordination between the two MCUs. Now, let us consider the four acceleration modes on which the generation of DNNs relies. 1) Extended Inference: in this acceleration mode, an extended version of the baseline DNN is generated by adding neurons to each layer that have the same dimensions as in the baseline DNN. This addition of neurons and connections in an embedded device without an increase in device size improves inference accuracy. 2) Expedited Inference: In this mode, the extended convolutional layers are connected to the main MCU ensuring minimal communication with the accelerator. Leveraging parallelism in the architecture cuts down the inference

time by 45 % expediting the process. 3) Ensembled Inference: Unlike the other two acceleration modes, this mode runs a separate DNN model on the accelerator by taking additional input from say sensors, to get a second DNN and combines the output of the two independent DNNs to boost inference accuracy. This mode works as a trade-off between accuracy and speedup but can lead to the development of multi-model, multi-objective sensing, and inference systems. 4) Latent training: This acceleration mode enables intermittent on-device training of the DNN for new data on the accelerator while allowing the main MCU to parallelly execute the inference tasks. While training large data with multiple parameters, the important points to consider are, that the data must be labeled, and the back-propagation algorithm consumes a lot of energy. To resolve the data labeling issue, Neuro.Zero system relies on an external, high accuracy inference system to obtain the labels at run-time and for back-propagation algorithm issues, new Energy-Aware acceleration algorithms are introduced namely Step-Up Inference and Skip-Out training. In Step-Up Inference, there is gradual addition of steps starting from the baseline DNN by training and adding new CNN filters to the previous step which enables higher inference accuracy based on the current energy level. The skip-Out training follows two algorithms, 1) Skip-Out Back-Propagation, where back-propagation of selected weights using Bernoulli distribution is skipped and removed by drop-out, 2) Skip-Out Feed-Forward, here instead of skipping, the average skip-out rate is applied to the activation of all neurons. These algorithms not only increase the training accuracy but diminish the over-fitting problem. Another algorithm in use for the four acceleration modes is the Adaptive-scale Fixed-Point Arithmetic, this algorithm uses Adaptive-scale Multiply-Accumulate operations to mitigate overflow and precision loss problems of Fixed-Point numbers.

Neuro.Zero has been practically applied to real-time systems to examine its functioning. It was tested in two scenarios, A) Traffic Sign Recognizer: a camera was fitted on Neuro.Zero hardware and almost 40000 images were captured and used as training and testing images to evaluate the performance of the accelerator in all four acceleration modes. Different modes have provided different outcomes providing an accuracy of approximately 85%, B) Voice Command Recognizer: a limited-vocabulary speech recognition with 10 voice commands and RF energy harvester is exploited to obtain frequency information from the MCU. Then adaptive-scale fixed-point algorithm of 16 bit-width for numerical operations was utilized to compare the frequency data. Testing Neuro.Zero at all acceleration modes gave the estimated execution speed of 21% less than baseline and accuracy up to 10% less than baseline DNN.

Conclusion

Reviewing the working, pros, and cons of the Neuro.Zero hardware we analyze and conclude that it guarantees sensing and inference for all sensor data with no trade-off accompanying higher run-time performance. Since Neuro.Zero is implemented with low-power, low-cost MCUs and its source code available to all the developers, it is the most affordable intelligent system to everyone. It has also been tested for the overhead of acceleration, intensely evaluated the applications, algorithms and proved to be efficient enough for regular usage. Though there is a scope for improvement in accuracy compared to the baseline DNN, it has carved a way for low-energy multi-modal sensing and inference system.